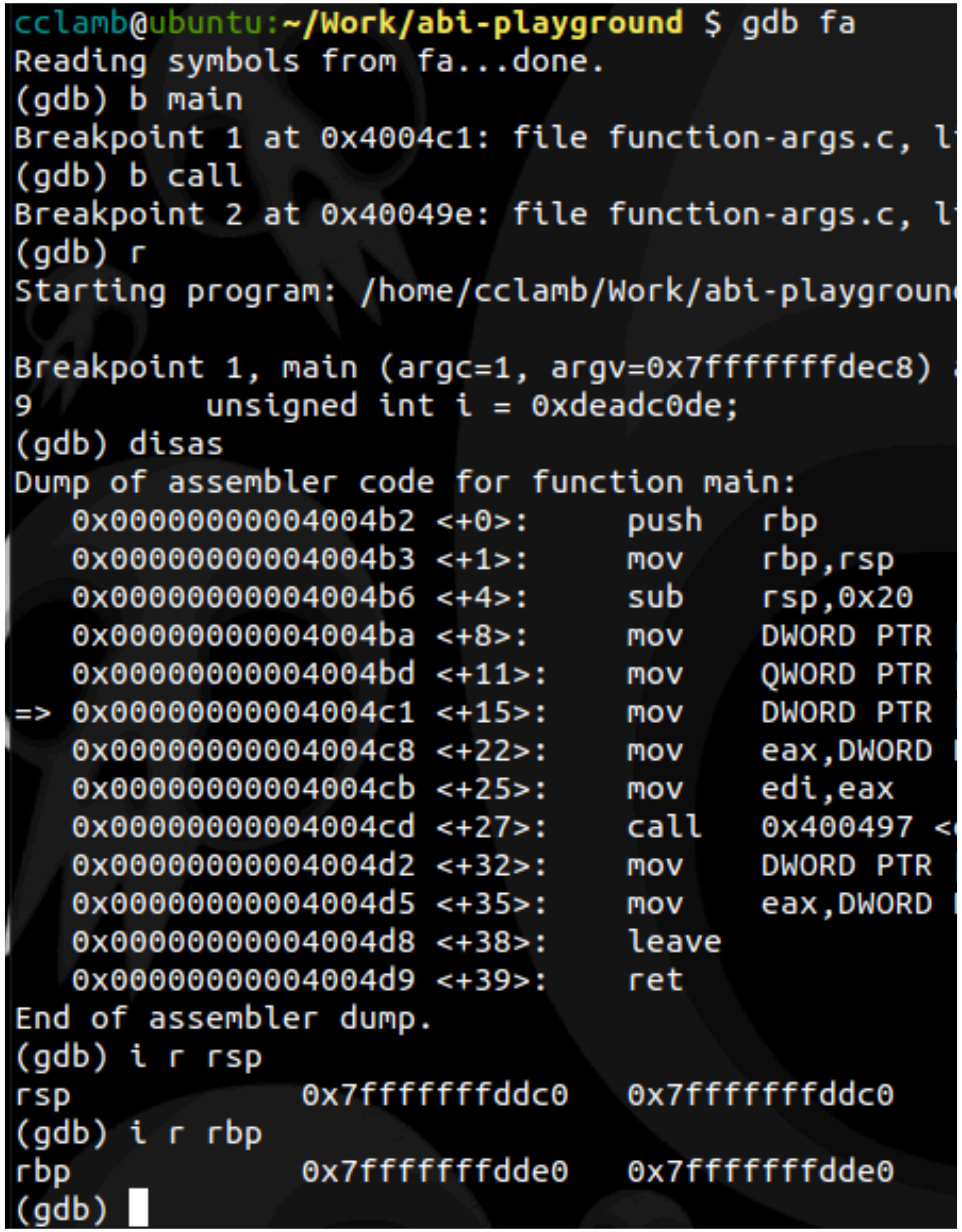
Technical Cybersecurity

Watching the Stack

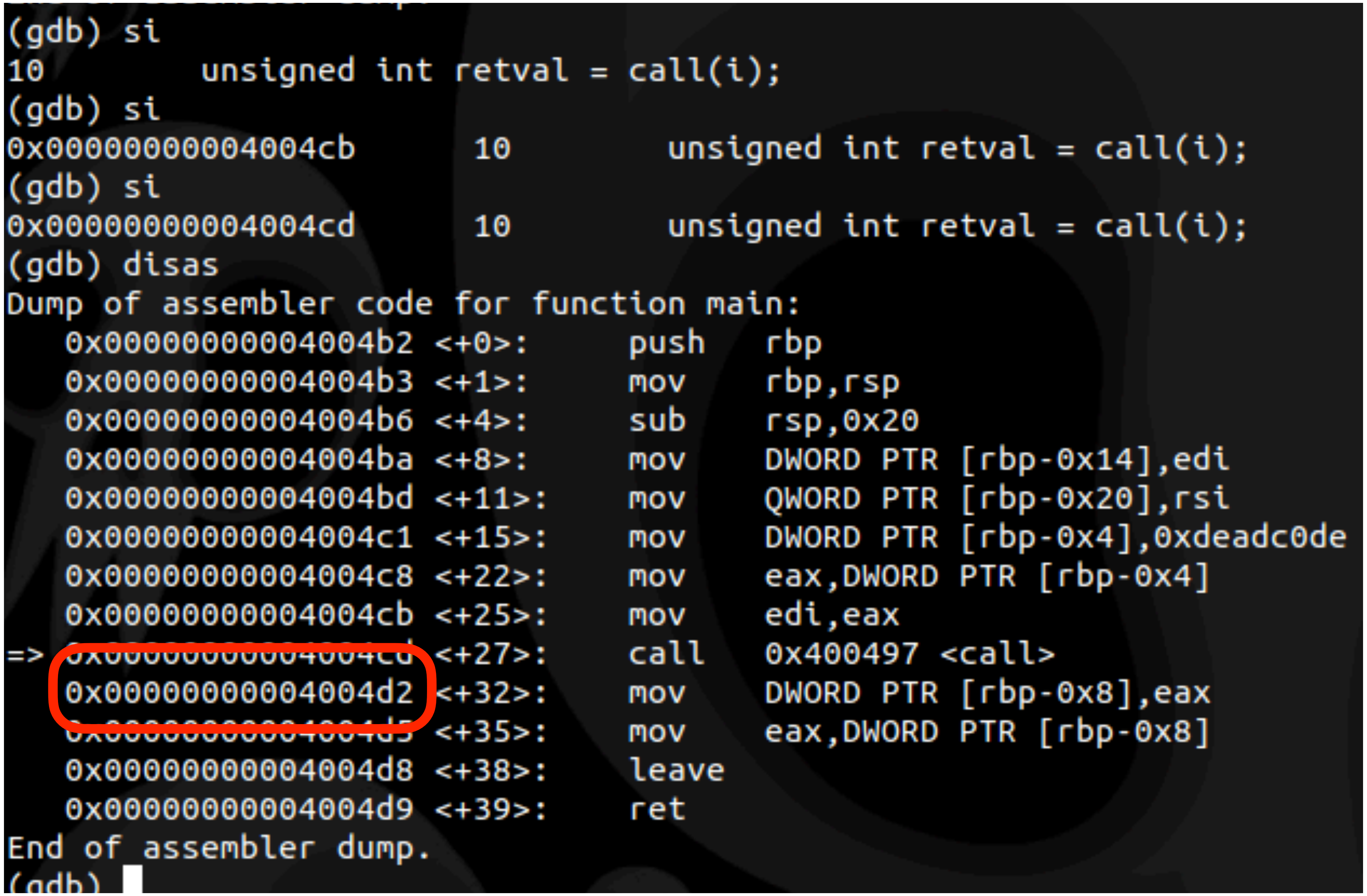
Back to **FA**

START UP **FA**

- Set a couple of breakpoints
- Run the program
- Let's check some stuff out!



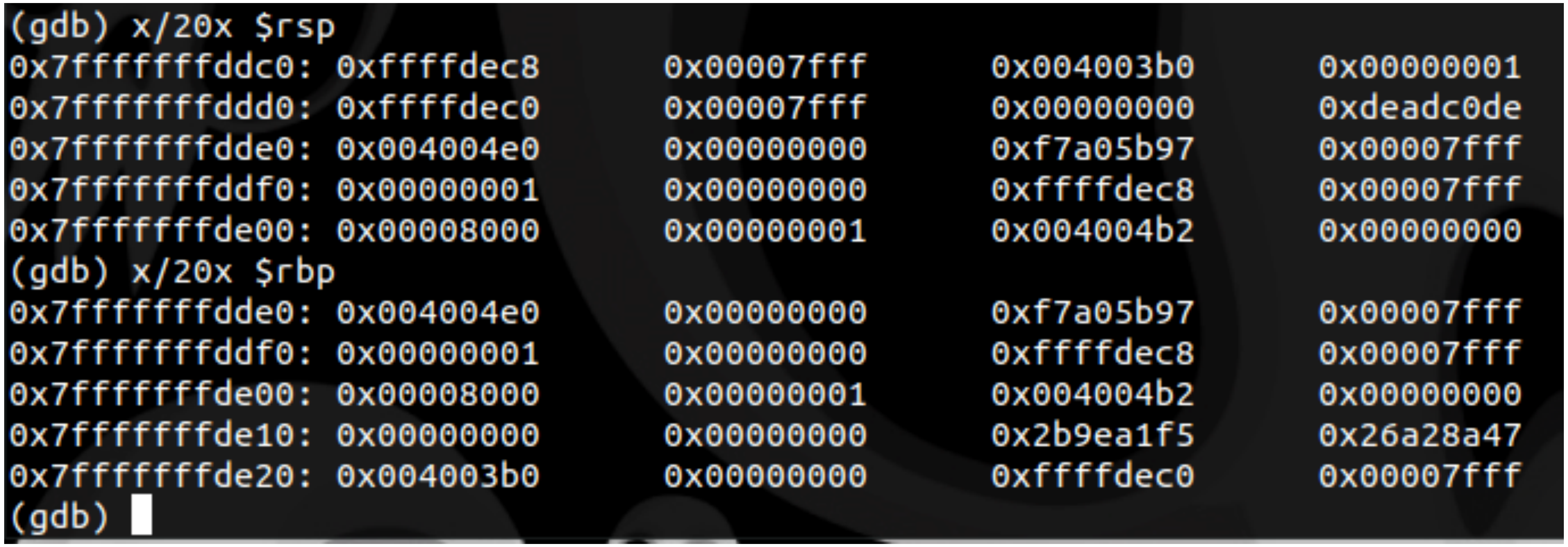
```
cclamb@ubuntu:~/Work/abi-playground $ gdb fa
Reading symbols from fa...done.
(gdb) b main
Breakpoint 1 at 0x4004c1: file function-args.c, li
(gdb) b call
Breakpoint 2 at 0x40049e: file function-args.c, li
(gdb) r
Starting program: /home/cclamb/Work/abi-playgroun

Breakpoint 1, main (argc=1, argv=0x7fffffffdec8)
9           unsigned int i = 0xdeadc0de;
(gdb) disas
Dump of assembler code for function main:
   0x00000000004004b2 <+0>:     push   rbp
   0x00000000004004b3 <+1>:     mov    rbp,rsp
   0x00000000004004b6 <+4>:     sub    rsp,0x20
   0x00000000004004ba <+8>:     mov    DWORD PTR
   0x00000000004004bd <+11>:    mov    QWORD PTR
=> 0x00000000004004c1 <+15>:    mov    DWORD PTR
   0x00000000004004c8 <+22>:    mov    eax,DWORD
   0x00000000004004cb <+25>:    mov    edi,eax
   0x00000000004004cd <+27>:    call   0x400497 <
   0x00000000004004d2 <+32>:    mov    DWORD PTR
   0x00000000004004d5 <+35>:    mov    eax,DWORD
   0x00000000004004d8 <+38>:    leave
   0x00000000004004d9 <+39>:    ret
End of assembler dump.
(gdb) i r rsp
rsp            0x7fffffffddc0   0x7fffffffddc0
(gdb) i r rbp
rbp            0x7fffffffdde0   0x7fffffffdde0
(gdb) 
```



```
End of assembler dump.
(gdb) si
10          unsigned int retval = call(i);
(gdb) si
0x00000000004004cb      10          unsigned int retval = call(i);
(gdb) si
0x00000000004004cd      10          unsigned int retval = call(i);
(gdb) disas
Dump of assembler code for function main:
   0x00000000004004b2 <+0>:     push   rbp
   0x00000000004004b3 <+1>:     mov    rbp,rsp
   0x00000000004004b6 <+4>:     sub    rsp,0x20
   0x00000000004004ba <+8>:     mov    DWORD PTR [rbp-0x14],edi
   0x00000000004004bd <+11>:    mov    QWORD PTR [rbp-0x20],rsi
   0x00000000004004c1 <+15>:    mov    DWORD PTR [rbp-0x4],0xdeadc0de
   0x00000000004004c8 <+22>:    mov    eax,DWORD PTR [rbp-0x4]
   0x00000000004004cb <+25>:    mov    edi,eax
=> 0x00000000004004cd <+27>:    call   0x400497 <call>
   0x00000000004004d2 <+32>:    mov    DWORD PTR [rbp-0x8],eax
   0x00000000004004d5 <+35>:    mov    eax,DWORD PTR [rbp-0x8]
   0x00000000004004d8 <+38>:    leave
   0x00000000004004d9 <+39>:    ret
End of assembler dump.
(gdb)
```

Step Lightly Now…

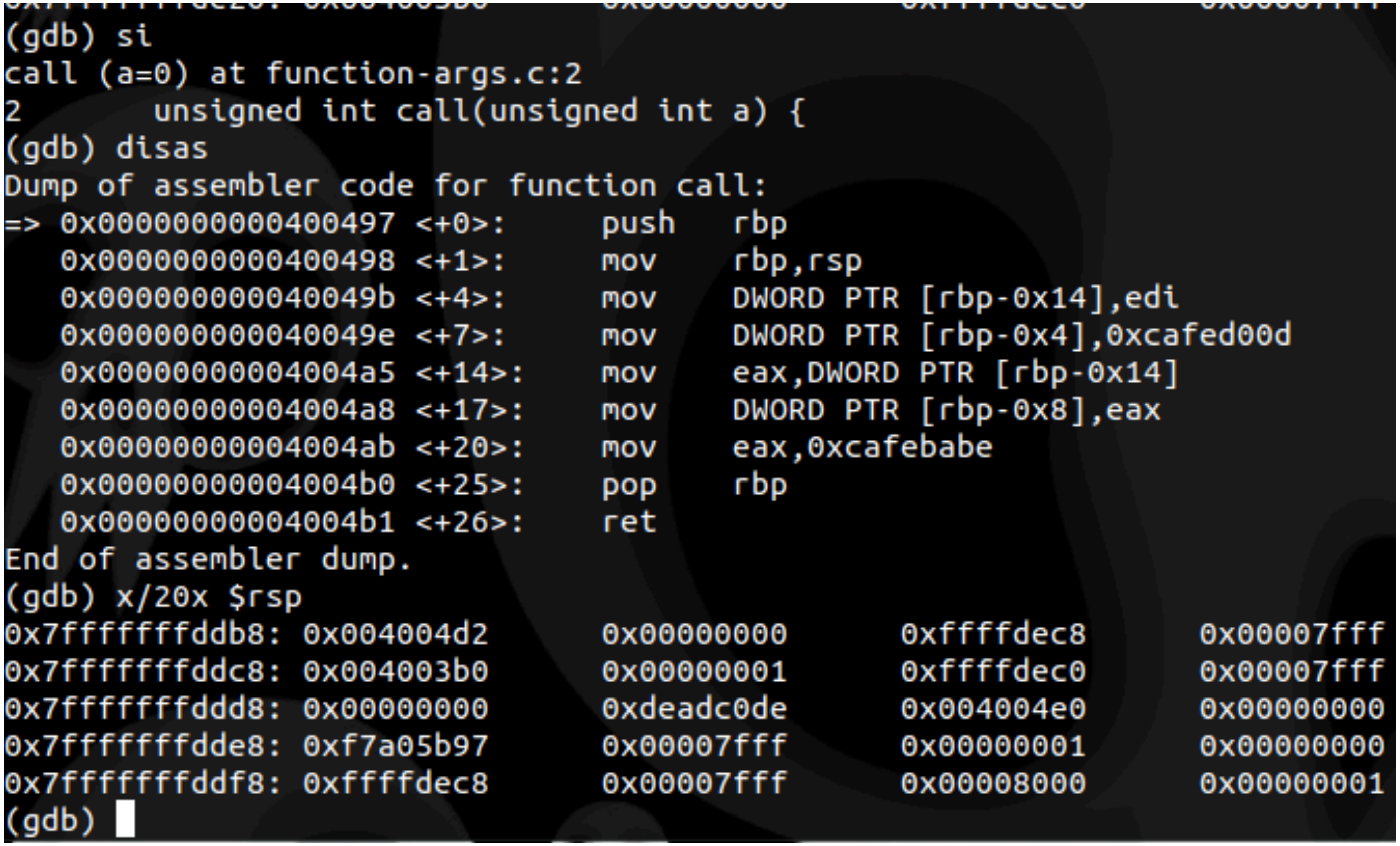
(Note the address after the **call**: 0x4004d2)



```
(gdb) x/20x $rsp
0x7fffffffddc0: 0xffffdec8      0x00007fff      0x004003b0      0x00000001
0x7fffffffddd0: 0xffffdec0      0x00007fff      0x00000000      0xdeadc0de
0x7fffffffdde0: 0x004004e0      0x00000000      0xf7a05b97      0x00007fff
0x7fffffffddf0: 0x00000001      0x00000000      0xffffdec8      0x00007fff
0x7fffffffde00: 0x00008000      0x00000001      0x004004b2      0x00000000
(gdb) x/20x $rbp
0x7fffffffdde0: 0x004004e0      0x00000000      0xf7a05b97      0x00007fff
0x7fffffffddf0: 0x00000001      0x00000000      0xffffdec8      0x00007fff
0x7fffffffde00: 0x00008000      0x00000001      0x004004b2      0x00000000
0x7fffffffde10: 0x00000000      0x00000000      0x2b9ea1f5      0x26a28a47
0x7fffffffde20: 0x004003b0      0x00000000      0xffffdec0      0x00007fff
(gdb) 
```

Memory Contents

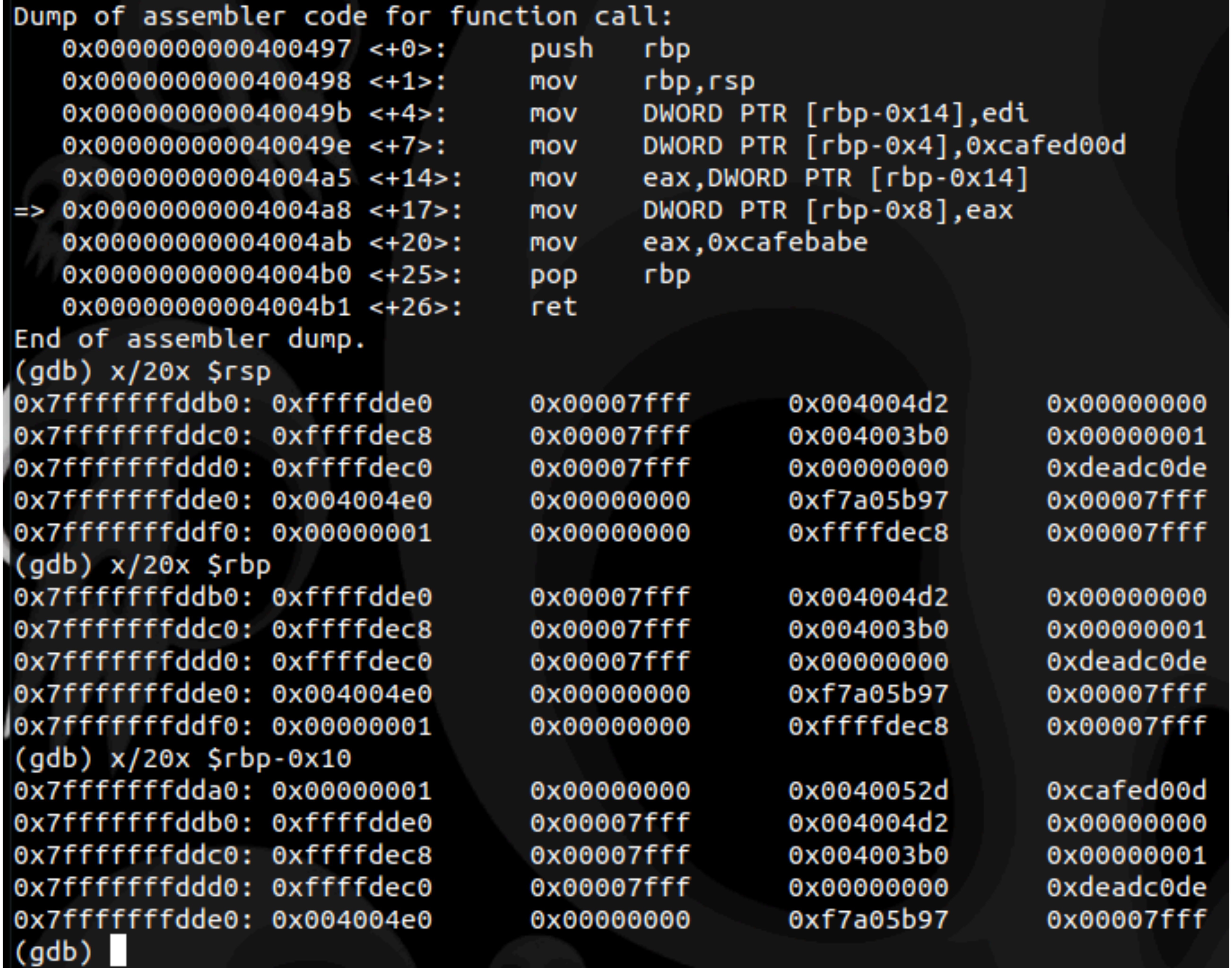
Examining the Stack



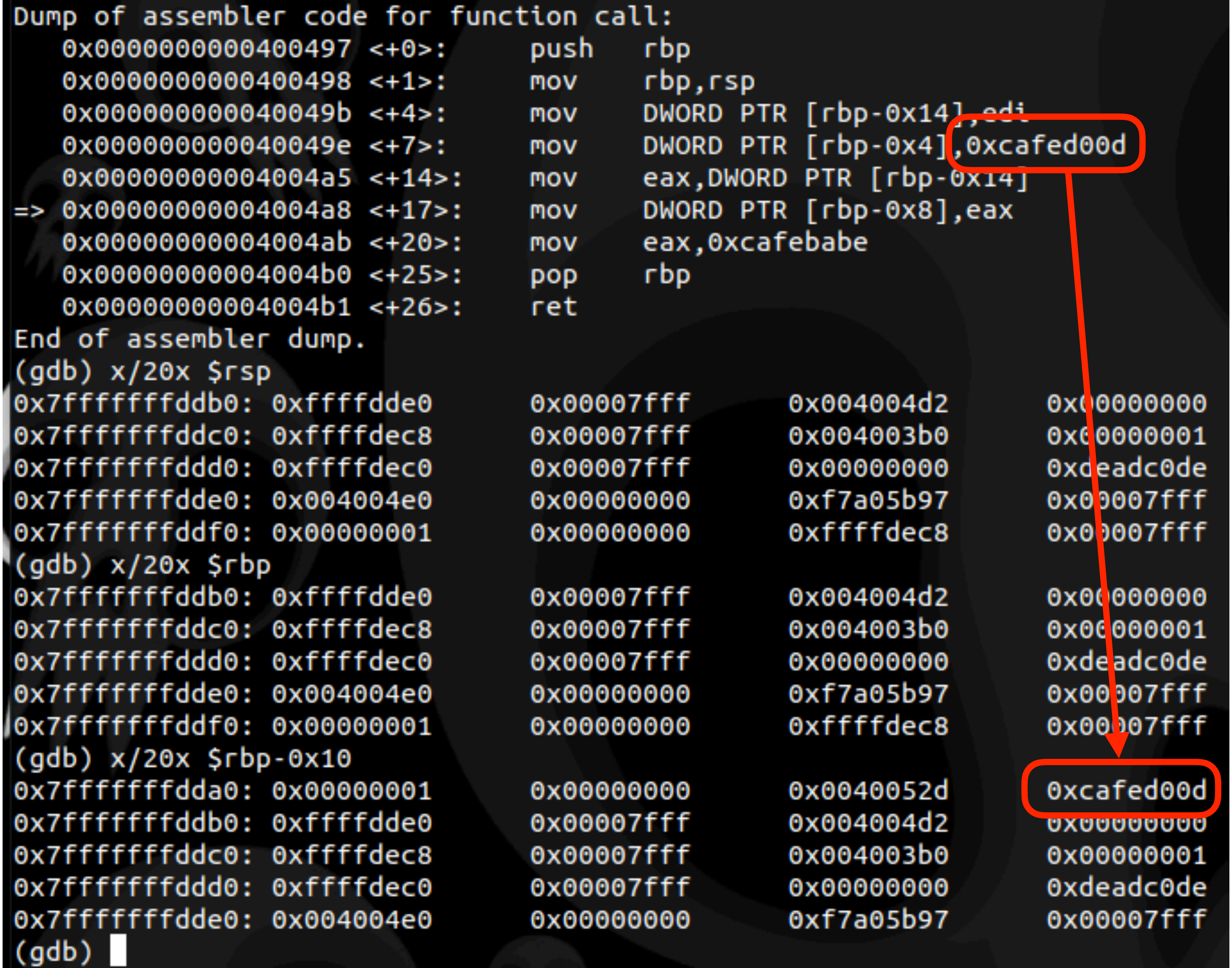
```
(gdb) si
call (a=0) at function-args.c:2
2       unsigned int call(unsigned int a) {
(gdb) disas
Dump of assembler code for function call:
=> 0x0000000000400497 <+0>:     push   rbp
   0x0000000000400498 <+1>:     mov    rbp,rsp
   0x000000000040049b <+4>:     mov    DWORD PTR [rbp-0x14],edi
   0x000000000040049e <+7>:     mov    DWORD PTR [rbp-0x4],0xcafed00d
   0x00000000004004a5 <+14>:    mov    eax,DWORD PTR [rbp-0x14]
   0x00000000004004a8 <+17>:    mov    DWORD PTR [rbp-0x8],eax
   0x00000000004004ab <+20>:    mov    eax,0xcafebabe
   0x00000000004004b0 <+25>:    pop    rbp
   0x00000000004004b1 <+26>:    ret
End of assembler dump.
(gdb) x/20x $rsp
0x7fffffffddb8: 0x004004d2      0x00000000      0xffffdec8      0x00007fff
0x7fffffffddc8: 0x004003b0      0x00000001      0xffffdec0      0x00007fff
0x7fffffffddd8: 0x00000000      0xdeadc0de      0x004004e0      0x00000000
0x7fffffffdde8: 0xf7a05b97      0x00007fff      0x00000001      0x00000000
0x7fffffffddf8: 0xffffdec8      0x00007fff      0x00008000      0x00000001
(gdb)
```

Stepping & Memory

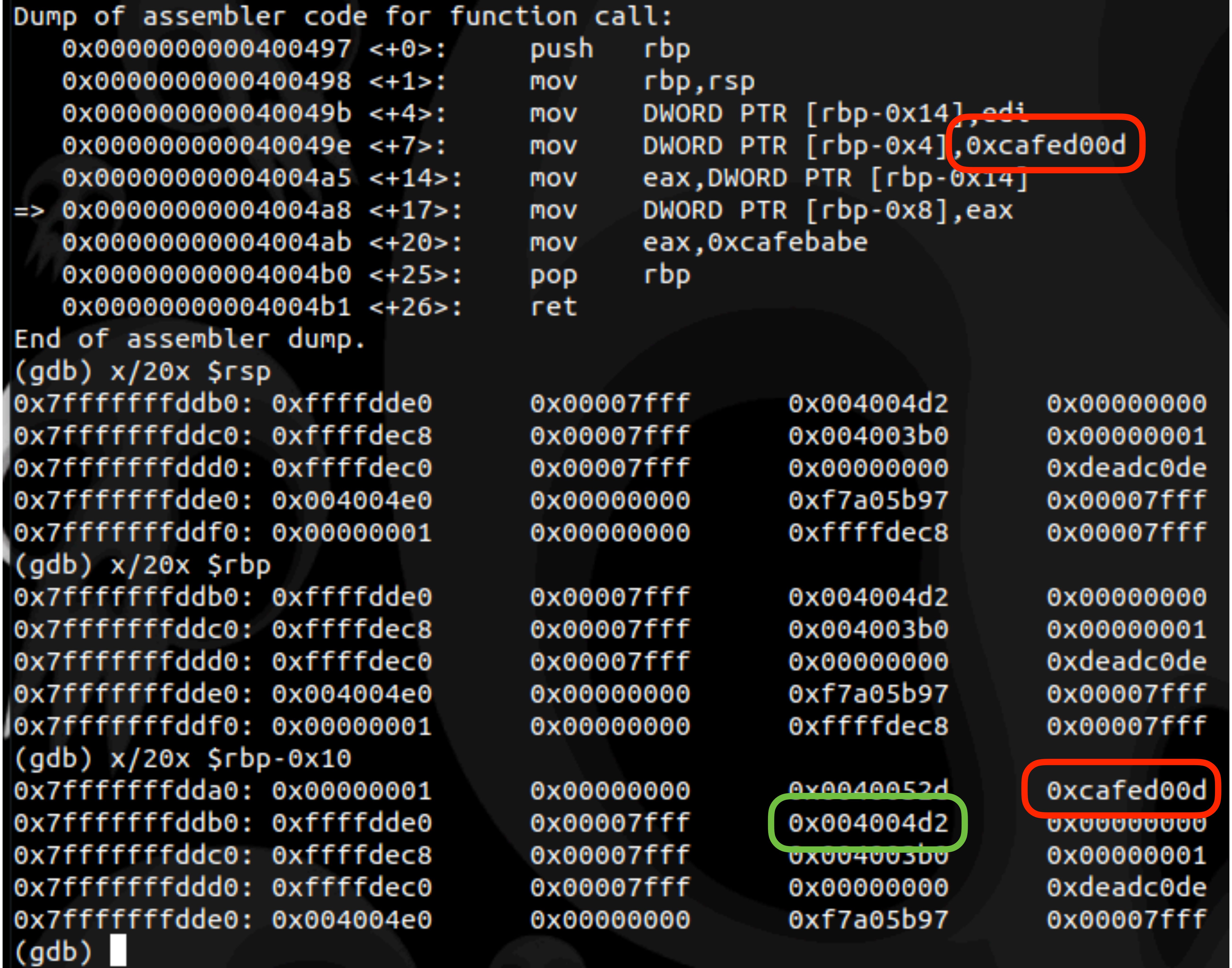
(Look at the contents at 0x7fffffffddb8; look familiar?)



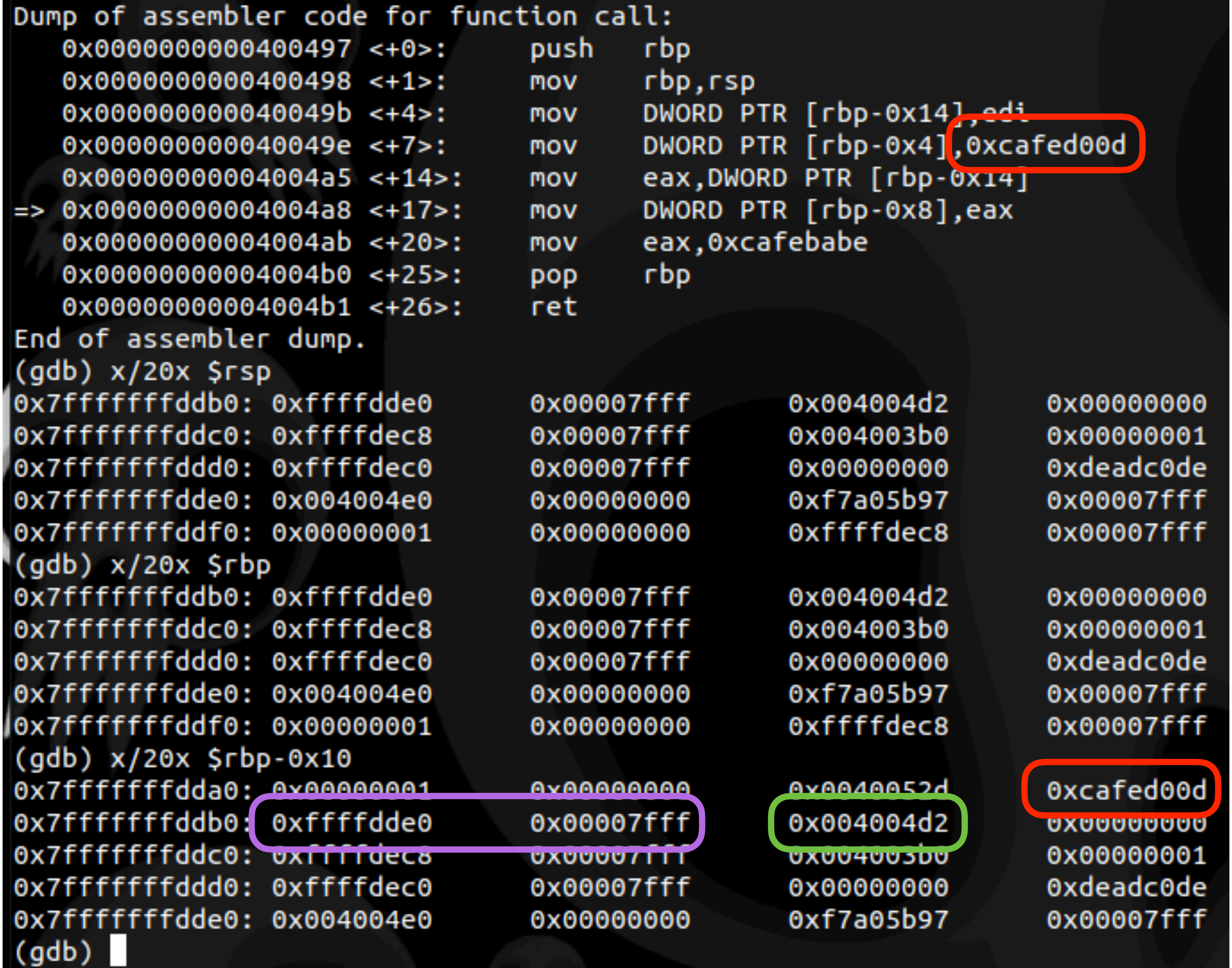
```
Dump of assembler code for function call:
   0x0000000000400497 <+0>:     push   rbp
   0x0000000000400498 <+1>:     mov    rbp,rsp
   0x000000000040049b <+4>:     mov    DWORD PTR [rbp-0x14],edi
   0x000000000040049e <+7>:     mov    DWORD PTR [rbp-0x4],0xcafed00d
   0x00000000004004a5 <+14>:    mov    eax,DWORD PTR [rbp-0x14]
=> 0x00000000004004a8 <+17>:    mov    DWORD PTR [rbp-0x8],eax
   0x00000000004004ab <+20>:    mov    eax,0xcafebabe
   0x00000000004004b0 <+25>:    pop    rbp
   0x00000000004004b1 <+26>:    ret
End of assembler dump.
(gdb) x/20x $rsp
0x7fffffffddb0: 0xffffdde0      0x00007fff      0x004004d2      0x00000000
0x7fffffffddc0: 0xffffdec8      0x00007fff      0x004003b0      0x00000001
0x7fffffffddd0: 0xffffdec0      0x00007fff      0x00000000      0xdeadc0de
0x7fffffffdde0: 0x004004e0      0x00000000      0xf7a05b97      0x00007fff
0x7fffffffddf0: 0x00000001      0x00000000      0xffffdec8      0x00007fff
(gdb) x/20x $rbp
0x7fffffffddb0: 0xffffdde0      0x00007fff      0x004004d2      0x00000000
0x7fffffffddc0: 0xffffdec8      0x00007fff      0x004003b0      0x00000001
0x7fffffffddd0: 0xffffdec0      0x00007fff      0x00000000      0xdeadc0de
0x7fffffffdde0: 0x004004e0      0x00000000      0xf7a05b97      0x00007fff
0x7fffffffddf0: 0x00000001      0x00000000      0xffffdec8      0x00007fff
(gdb) x/20x $rbp-0x10
0x7fffffffdda0: 0x00000001      0x00000000      0x0040052d      0xcafed00d
0x7fffffffddb0: 0xffffdde0      0x00007fff      0x004004d2      0x00000000
0x7fffffffddc0: 0xffffdec8      0x00007fff      0x004003b0      0x00000001
0x7fffffffddd0: 0xffffdec0      0x00007fff      0x00000000      0xdeadc0de
0x7fffffffdde0: 0x004004e0      0x00000000      0xf7a05b97      0x00007fff
(gdb)
```



```
Dump of assembler code for function call:
   0x0000000000400497 <+0>:     push   rbp
   0x0000000000400498 <+1>:     mov    rbp,rsp
   0x000000000040049b <+4>:     mov    DWORD PTR [rbp-0x14],edi
   0x000000000040049e <+7>:     mov    DWORD PTR [rbp-0x4],0xcafed00d
   0x00000000004004a5 <+14>:    mov    eax,DWORD PTR [rbp-0x14]
=> 0x00000000004004a8 <+17>:    mov    DWORD PTR [rbp-0x8],eax
   0x00000000004004ab <+20>:    mov    eax,0xcafebabe
   0x00000000004004b0 <+25>:    pop    rbp
   0x00000000004004b1 <+26>:    ret
End of assembler dump.
(gdb) x/20x $rsp
0x7fffffffddb0: 0xffffdde0      0x00007fff      0x004004d2      0x00000000
0x7fffffffddc0: 0xffffdec8      0x00007fff      0x004003b0      0x00000001
0x7fffffffddd0: 0xffffdec0      0x00007fff      0x00000000      0xdeadc0de
0x7fffffffdde0: 0x004004e0      0x00000000      0xf7a05b97      0x00007fff
0x7fffffffddf0: 0x00000001      0x00000000      0xffffdec8      0x00007fff
(gdb) x/20x $rbp
0x7fffffffddb0: 0xffffdde0      0x00007fff      0x004004d2      0x00000000
0x7fffffffddc0: 0xffffdec8      0x00007fff      0x004003b0      0x00000001
0x7fffffffddd0: 0xffffdec0      0x00007fff      0x00000000      0xdeadc0de
0x7fffffffdde0: 0x004004e0      0x00000000      0xf7a05b97      0x00007fff
0x7fffffffddf0: 0x00000001      0x00000000      0xffffdec8      0x00007fff
(gdb) x/20x $rbp-0x10
0x7fffffffdda0: 0x00000001      0x00000000      0x0040052d      0xcafed00d
0x7fffffffddb0: 0xffffdde0      0x00007fff      0x004004d2      0x00000000
0x7fffffffddc0: 0xffffdec8      0x00007fff      0x004003b0      0x00000001
0x7fffffffddd0: 0xffffdec0      0x00007fff      0x00000000      0xdeadc0de
0x7fffffffdde0: 0x004004e0      0x00000000      0xf7a05b97      0x00007fff
(gdb) 
```



```
Dump of assembler code for function call:
   0x0000000000400497 <+0>:     push   rbp
   0x0000000000400498 <+1>:     mov    rbp,rsp
   0x000000000040049b <+4>:     mov    DWORD PTR [rbp-0x14],edi
   0x000000000040049e <+7>:     mov    DWORD PTR [rbp-0x4],0xcafed00d
   0x00000000004004a5 <+14>:    mov    eax,DWORD PTR [rbp-0x14]
=> 0x00000000004004a8 <+17>:    mov    DWORD PTR [rbp-0x8],eax
   0x00000000004004ab <+20>:    mov    eax,0xcafebabe
   0x00000000004004b0 <+25>:    pop    rbp
   0x00000000004004b1 <+26>:    ret
End of assembler dump.
(gdb) x/20x $rsp
0x7fffffffddb0: 0xffffdde0      0x00007fff      0x004004d2      0x00000000
0x7fffffffddc0: 0xffffdec8      0x00007fff      0x004003b0      0x00000001
0x7fffffffddd0: 0xffffdec0      0x00007fff      0x00000000      0xdeadc0de
0x7fffffffdde0: 0x004004e0      0x00000000      0xf7a05b97      0x00007fff
0x7fffffffddf0: 0x00000001      0x00000000      0xffffdec8      0x00007fff
(gdb) x/20x $rbp
0x7fffffffddb0: 0xffffdde0      0x00007fff      0x004004d2      0x00000000
0x7fffffffddc0: 0xffffdec8      0x00007fff      0x004003b0      0x00000001
0x7fffffffddd0: 0xffffdec0      0x00007fff      0x00000000      0xdeadc0de
0x7fffffffdde0: 0x004004e0      0x00000000      0xf7a05b97      0x00007fff
0x7fffffffddf0: 0x00000001      0x00000000      0xffffdec8      0x00007fff
(gdb) x/20x $rbp-0x10
0x7fffffffdda0: 0x00000001      0x00000000      0x0040052d      0xcafed00d
0x7fffffffddb0: 0xffffdde0      0x00007fff      0x004004d2      0x00000000
0x7fffffffddc0: 0xffffdec8      0x00007fff      0x004003b0      0x00000001
0x7fffffffddd0: 0xffffdec0      0x00007fff      0x00000000      0xdeadc0de
0x7fffffffdde0: 0x004004e0      0x00000000      0xf7a05b97      0x00007fff
(gdb)
```



```
Dump of assembler code for function call:
   0x0000000000400497 <+0>:     push   rbp
   0x0000000000400498 <+1>:     mov    rbp,rsp
   0x000000000040049b <+4>:     mov    DWORD PTR [rbp-0x14],edi
   0x000000000040049e <+7>:     mov    DWORD PTR [rbp-0x4],0xcafed00d
   0x00000000004004a5 <+14>:    mov    eax,DWORD PTR [rbp-0x14]
=> 0x00000000004004a8 <+17>:    mov    DWORD PTR [rbp-0x8],eax
   0x00000000004004ab <+20>:    mov    eax,0xcafebabe
   0x00000000004004b0 <+25>:    pop    rbp
   0x00000000004004b1 <+26>:    ret
End of assembler dump.
(gdb) x/20x $rsp
0x7fffffffddb0: 0xffffdde0      0x00007fff      0x004004d2      0x00000000
0x7fffffffddc0: 0xffffdec8      0x00007fff      0x004003b0      0x00000001
0x7fffffffddd0: 0xffffdec0      0x00007fff      0x00000000      0xdeadc0de
0x7fffffffdde0: 0x004004e0      0x00000000      0xf7a05b97      0x00007fff
0x7fffffffddf0: 0x00000001      0x00000000      0xffffdec8      0x00007fff
(gdb) x/20x $rbp
0x7fffffffddb0: 0xffffdde0      0x00007fff      0x004004d2      0x00000000
0x7fffffffddc0: 0xffffdec8      0x00007fff      0x004003b0      0x00000001
0x7fffffffddd0: 0xffffdec0      0x00007fff      0x00000000      0xdeadc0de
0x7fffffffdde0: 0x004004e0      0x00000000      0xf7a05b97      0x00007fff
0x7fffffffddf0: 0x00000001      0x00000000      0xffffdec8      0x00007fff
(gdb) x/20x $rbp-0x10
0x7fffffffdda0: 0x00000001      0x00000000      0x0040052d      0xcafed00d
0x7fffffffddb0: 0xffffdde0      0x00007fff      0x004004d2      0x00000000
0x7fffffffddc0: 0xffffdec8      0x00007fff      0x004003b0      0x00000001
0x7fffffffddd0: 0xffffdec0      0x00007fff      0x00000000      0xdeadc0de
0x7fffffffdde0: 0x004004e0      0x00000000      0xf7a05b97      0x00007fff
(gdb) █
```

Buffer Overflow!